Markov Chain Monte Carlo on Asymmetric GARCH Model Using the Adaptive Construction Scheme

Tetsuya Takaishi

Hiroshima University of Economics,
731-0192 Hiroshima, Japan
E-mail: takaishi@hiroshima-u.ac.jp

Abstract. We perform Markov chain Monte Carlo simulations for a Bayesian inference of the GJR-GARCH model which is one of asymmetric GARCH models. The adaptive construction scheme is used for the construction of the proposal density in the Metropolis-Hastings algorithm and the parameters of the proposal density are determined adaptively by using the data sampled by the Markov chain Monte Carlo simulation. We study the performance of the scheme with the artificial GJR-GARCH data. We find that the adaptive construction scheme samples GJR-GARCH parameters effectively and conclude that the Metropolis-Hastings algorithm with the adaptive construction scheme is an efficient method to the Bayesian inference of the GJR-GARCH model.

Key words: Markov Chain Monte Carlo, Bayesian inference, GJR-GARCH model, Metropolis-Hasting algorithm

1 Introduction

Price returns of financial assets such as stock indexes, exchange rates show various interesting properties which can not be derived from a simple assumption that the price returns follow the geometric Brownian motion. Those properties are now classified as stylized facts[1,2]. Typical examples of the stylized facts are (i) fat-tailed distribution of return (ii) volatility clustering (iii) slow decay of the autocorrelation time of the absolute returns.

In empirical finance the volatility is an important quantity to measure risk. In order to forecast future volatility it is important to make models which mimic the properties of the volatility. The most successful model is the Generalized Autoregressive Conditional Heteroscedasticity (GARCH) model by Engle[3] and Bollerslev[4], which can capture the property of volatility clustering and shows fat-tailed distributions.

In the original GARCH model[3,4], the volatility process is symmetric under positive and negative observations. However the volatility often shows higher response against negative shocks, which was first observed by Black[5]. This phenomenon is called the leverage effect. In order to incorporate this asymmetry effect into the model, some extended GARCH models have been proposed[6,7,8,9,10,11].

In this study we use the GJR-GARCH model by Glosten *et al.*[7] which introduces the asymmetry into the volatility process by an artificial indicator function.

In order to infer model parameters from financial data we employ Markov Chain Monte Carlo (MCMC) methods based on the Bayesian inference. Since there is no unique way to implement the MCMC scheme various MCMC methods for the GARCH models have been proposed[12,13,14,15,16,17,18,19]. In a survey on the MCMC methods of the GARCH models[17] it is shown that Acceptance-Rejection/ Metropolis-Hastings (AR/MH) algorithm works better than other algorithms. In the AR/MH algorithm the proposal density is assumed to be a multivariate Student's t-distribution and the parameters to specify the distribution are estimated by the Maximum Likelihood (ML) technique. Recently an alternative method to estimate those parameters without relying on the ML technique was proposed [20,21,22]. In the method the parameters are determined by using the data generated by an MCMC method and updated adaptively during the MCMC simulation. We call this method "adaptive construction scheme".

The adaptive construction scheme was tested for GARCH and QGARCH models[20,21,22] and it is shown that the adaptive construction scheme can significantly reduce the correlation between sampled data. In this study we apply the adaptive construction scheme to the GJR-GARCH model and study the efficiency of the adaptive construction scheme.

2   GJR-GARCH Model

The GJR-GARCH model[7] is given by

$$y_t = \sigma_t \epsilon_t, \tag{1}$$

$$\sigma_t^2 = \omega + \alpha y_{t-1}^2 + \lambda \{1_{y_{t-1}<0}\} y_{t-1}^2 + \beta \sigma_{t-1}^2, \tag{2}$$

where ϵ_t is an independent normal error $\sim N(0,1)$ and y_t are observations. The indicator function $1_{y_{t-1}<0}$ is 1 if $y_{t-1} < 0$, and otherwise 0. This indicator function introduces the asymmetry in the time series and it generates higher volatilities after negative shocks than positive ones.

3   Bayesian inference

Using the Bayes' rule the posterior density $\pi(\theta|y)$ with n observations $y = (y_1, y_2, \ldots, y_n)$ is given by

$$\pi(\theta|y) \propto L(y|\theta)\pi(\theta), \tag{3}$$

where $L(y|\theta)$ is the likelihood function. $\pi(\theta)$ is the prior density for θ. The functional form of $\pi(\theta)$ is not known a priori. In this study we assume that the prior density $\pi(\theta)$ is constant.

The likelihood function for the GJR-GARCH model is given by

$$L(y|\theta) = \Pi_{i=1}^n \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp{(-\frac{y_t^2}{2\sigma_t^2})}, \tag{4}$$

where $\theta = (\alpha, \beta, \omega, \lambda)$ stands for the GJR-GARCH parameters. σ_t^2 is given by eq.(2).

Using $\pi(\theta|y)$ the GJR-GARCH parameters are inferred as the expectation values given by

$$\langle \theta \rangle = \frac{1}{Z} \int \theta \pi(\theta|y) d\theta, \tag{5}$$

where $Z = \int \pi(\theta|y) d\theta$ is a normalization constant irrelevant to MCMC estimations.

3.1 Markov Chain Monte Carlo

In general it is difficult to evaluate eq.(5) analytically. The MCMC technique gives a method to estimate eq.(5) numerically. The basic procedure of the MCMC method is as follows. First we sample θ drawn from a probability distribution $\pi(\theta|y)$. Sampling is done by a technique which produces a Markov chain. After sampling some data, we evaluate the expectation value as an average value over the sampled data $\theta^{(i)}$,

$$\langle \theta \rangle = \lim_{k \to \infty} \frac{1}{k} \sum_{i=1}^{k} \theta^{(i)}, \tag{6}$$

where k is the number of the sampled data. The statistical error for k independent data is proportional to $\frac{1}{\sqrt{k}}$. In general, however, the data generated by the MCMC process are correlated. As a result the statistical error will be proportional to $\sqrt{\frac{2\tau}{k}}$ where τ is the autocorrelation time among the sampled data.

3.2 Metropolis-Hastings algorithm

The MH algorithm[24] is a generalized version of the Metropolis algorithm[23]. Let us consider to generate data x from a probability distribution $P(x)$. The MH algorithm consists of the following steps. First starting from x, we propose a candidate x' which is drawn from a certain probability distribution $g(x'|x)$ which we call proposal density. Then we accept the candidate x' with a probability $P_{MH}(x, x')$ as the next value of the Markov chain:

$$P_{MH}(x, x') = \min\left[1, \frac{P(x')}{P(x)} \frac{g(x|x')}{g(x'|x)}\right]. \tag{7}$$

If x' is rejected we keep the previous value x. We repeat these steps.

When the proposal density does not depend on the previous value, i.e. $g(x|x') = g(x)$ we obtain

$$P_{MH}(x, x') = \min\left[1, \frac{P(x')}{P(x)} \frac{g(x)}{g(x')}\right]. \tag{8}$$

4   Adaptive construction scheme

The efficiency of the MH algorithm depends on how we choose the proposal density. By choosing an adequate proposal density for the MH algorithm one can reduce the correlation between the sampled data. The posterior density of GARCH parameters often resembles to a Gaussian-like shape. Thus one may choose a density similar to a Gaussian distribution as the proposal density. Such attempts have been done by Mitsui, Watanabe[16] and Asai[17]. They used a multivariate Student's t-distribution in order to cover the tails of the posterior density and determined the parameters to specify the distribution by using the ML technique. Here we also use a multivariate Student's t-distribution but determine the parameters through MCMC simulations.

The (p-dimensional) multivariate Student's t-distribution is given by

$$g(\theta) = \frac{\Gamma((\nu + p)/2)/\Gamma(\nu/2)}{\det \Sigma^{1/2}(\nu\pi)^{p/2}}$$
$$\times \left[1 + \frac{(\theta - M)^t \Sigma^{-1}(\theta - M)}{\nu}\right]^{-(\nu+p)/2}, \tag{9}$$

where θ and M are column vectors,

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_p \end{bmatrix}, M = \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_p \end{bmatrix}, \tag{10}$$

and $M_i = E(\theta_i)$. Σ is the covariance matrix defined as

$$\frac{\nu\Sigma}{\nu - 2} = E[(\theta - M)(\theta - M)^t]. \tag{11}$$

ν is a parameter to tune the shape of Student's t-distribution. When $\nu \to \infty$ the Student's t-distribution goes to a Gaussian distribution. At small ν Student's t-distribution has a fat-tail. Since eq.(9) is independent of the previous value of θ, eq.(8) is used in the MH algorithm.

For the GJR-GARCH model, $p = 4$, $\theta = (\theta_1, \theta_2, \theta_3, \theta_4) = (\alpha, \beta, \omega, \lambda)$, and Σ is a 4×4 matrix. The unknown parameters in M and Σ are determined by using the data obtained from MCMC simulations. First we make a short run by the Metropolis algorithm and accumulate some data. Then we estimate M and Σ. Note that there is no need to estimate M and Σ accurately. Second we perform an MH simulation with the proposal density of eq.(9) with the estimated M and Σ. After accumulating more data, we recalculate M and Σ, and update M and Σ of eq.(9). By doing this, we adaptively change the shape of eq.(9) to fit the posterior density.

5   Numerical Study

In this section we study the performance of the adaptive construction scheme by using artificial GJR-GARCH data generated with known parameters. The GJR-GARCH parameters are set to $\alpha = 0.03$, $\beta = 0.85$, $\omega = 0.05$ and $\lambda = 0.1$. We have generated 2000 data by the GJR-GARCH process with these parameters. Fig.1 shows the time series of the 2000 data.
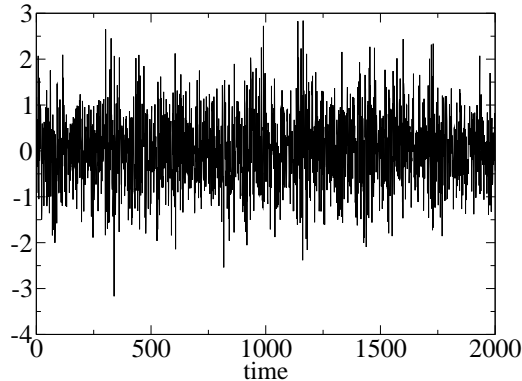
Fig. 1. The artificial times series generated by the GJR-GARCH process with $\alpha = 0.03$, $\beta = 0.85$, $\omega = 0.05$ and $\lambda = 0.1$.
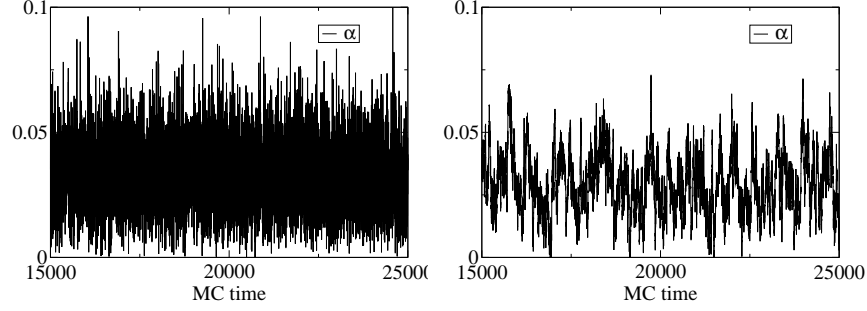
The MCMC method with the adaptive construction scheme is implemented as follows. First we make a short run by a standard Metropolis algorithm. The first 5000 data are discarded as burn-in process or thermalization. Then we accumulate 1000 data for M and Σ estimations. The estimated M and Σ are substituted to $g(\theta)$. In this study we take $\nu = 10$. We re-start a run by the MH algorithm with $g(\theta)$. Every 1000 updates we re-calculate M and Σ and update $g(\theta)$. We accumulate 100000 data for analysis.

For comparison we also use a standard Metropolis algorithm to infer the GJR-GARCH parameters and accumulate 100000 data for analysis. In this study the Metropolis algorithm is implemented as follows. We draw a candidate θ' by adding a small random value $\delta\theta$ to the present value θ:

$$\theta' = \theta + \delta\theta, \tag{12}$$

where $\delta\theta = d(r - 0.5)$. r is a uniform random number in $[0, 1]$ and d is a constant to tune the Metropolis acceptance. We choose d so that the acceptance becomes greater than 50%.

The results of the parameters inferred by the MH algorithm with the adaptive construction scheme and the Metropolis algorithm are summarized in Table 1. We see that both algorithms well reproduce the values of the input parameters within the standard deviation. Furthermore the obtained values and the standard

Fig. 2. Monte Carlo history of α from the MH algorithm with the adaptive construction scheme(left) and the Metropolis algorithm(right).

Table 1. Results of GJR-GARCH parameters. SD and SE stand for standard deviation and statistical error respectively.

	α	β	ω	λ
true	0.03	0.85	0.05	0.1
Adaptive	0.03285	0.85540	0.04522	0.08719
SD	0.0015	0.040	0.019	0.026
SE	0.00011	0.00025	0.00011	0.00066
2τ	2.8 ± 0.2	3.3 ± 0.7	4.6 ± 0.8	2.6 ± 0.2
Metropolis	0.0323	0.855	0.0454	0.0895
SD	0.0015	0.038	0.018	0.026
SE	0.0007	0.004	0.0018	0.0015
2τ	320 ± 100	1050 ± 350	990 ± 330	350 ± 110

deviations from both algorithms coincide well each other. This observation is not surprising because both algorithms are performed using the same artificial data and thus the posterior density is the same for both.

Fig.2 shows Monte Carlo time histories of the sampled α from the adaptive construction scheme (left) and Metropolis algorithm(right). We see that the data sampled by the Metropolis algorithm are substantially correlated. The similar behavior is also seen for the sampled data for other parameters.
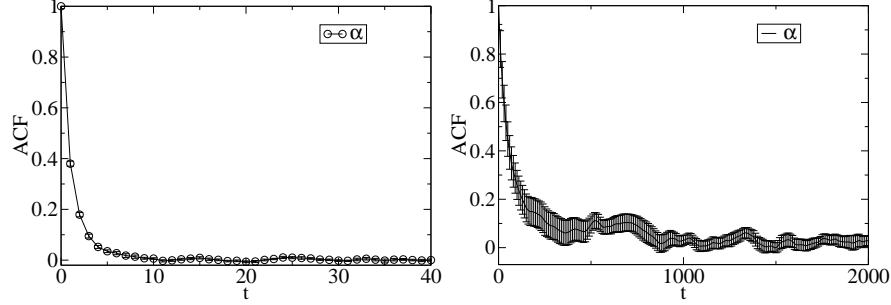
The correlations between the data can be measured by the autocorrelation function (ACF). The ACF of certain successive data $\theta^{(i)}$ is defined by

$$ACF(t) = \frac{\frac{1}{N}\sum_{j=1}^{N}(\theta^{(j)} - \langle\theta\rangle)(\theta^{(j+t)} - \langle\theta\rangle)}{\sigma_\theta^2}, \tag{13}$$

where $\langle\theta\rangle$ and σ_θ^2 are the average value and the variance of θ respectively.

Fig.3 compares the ACF of α sampled from the adaptive construction scheme(left) and the Metropolis algorithm(rigth). The ACF of the adaptive construction scheme decreases quickly as Monte Carlo time increases. On the other hand

the ACF of the Metropolis algorithm decreases very slowly, which indicates that the correlation between the sampled data is very large.

Fig. 3. Autocorrelation function of α sampled by the adaptive construction scheme(left) and the Metropolis algorithm(right).

To quantify the correlation between the data we calculate the autocorrelation time (ACT) τ defined by

$$\tau = \frac{1}{2} + \sum_{i=1}^{\infty} ACF(i). \tag{14}$$

2τ is also called "inefficiency factor". Results of τ are summarized in Table 1. We find that the ACT from the adaptive construction scheme have much smaller τ than those from the Metropolis simulations. For instance τ of α parameter from the adaptive construction scheme is decreased by a factor of 100 compared to that from the Metropolis algorithm. Furthermore the values of τ are about $2-3$ which are similar to the results of the AR/MH algorithm[16,17]. These results prove that the adaptive construction scheme is an efficient algorithm for sampling de-correlated data. The differences in τ also explain that the statistical errors from the adaptive construction scheme are much smaller than those from the Metropolis simulations.

Fig.4 shows the convergence property of the diagonal and off-diagonal elements of the covariance matrix V. Here V is defined by $V = E[(\theta - M)(\theta - M)^t]$. We see that all the diagonal elements of V quickly converge to certain values as the simulations are proceeded.

Fig.5 shows the acceptance at the MH algorithm with the adaptive proposal density of eq.(9). Each acceptance is calculated every 1000 updates and the calculation of the acceptance is based on the latest 1000 data. Surprisingly at the first stage of the simulation the acceptance already reaches a plateaus of about 70%. This indicates that the accuracy of the parameters of the proposal density are high enough for the MH algorithm already at the first stage.
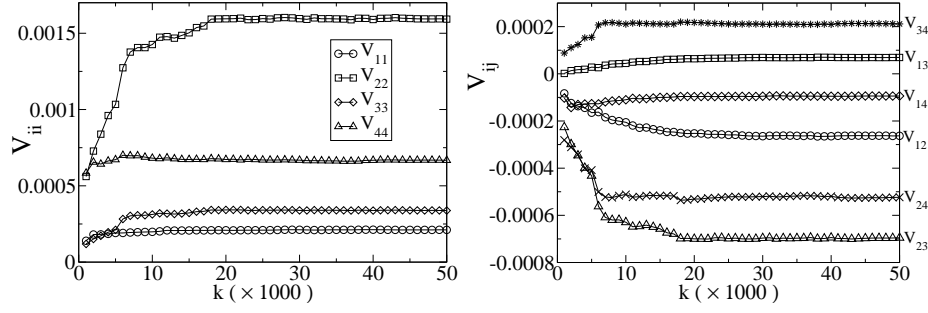
Fig. 4. The diagonal(left) and off-diagonal(right) elements of V as a function of the data size.
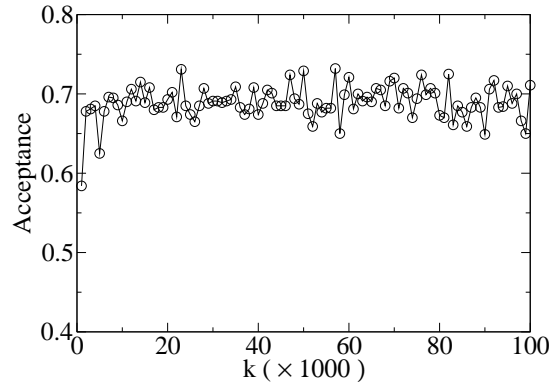
Fig. 5. Acceptance at MH step with the adaptive proposal density.

6   Conclusions

We have performed the MCMC simulations of the Bayesian inference on the GJR-GARCH model which is one of asymmetric GARCH models. The MCMC was implemented by the MH algorithm with the Student's t-distribution and the parameters of the Student's t-distribution were updated adaptively during the simulations. The autocorrelation times of the data sampled by the adaptive construction scheme are found to be very small. The obtained values of τ are similar to those of the AR/MH algorithm in [16,17]. Thus the adaptive construction scheme has the similar efficiency with the AR/MH algorithm. It is concluded that the adaptive construction scheme is also an efficient MCMC technique for the Bayesian inference of the GJR-GARCH model. The previous studies on the GARCH and QGARCH models[20,21,22] also reached the same conclusion.

It is also found that the acceptance of the MH algorithm quickly reaches a plateau of about 70% at the beginning of the simulation. This indicates that the parameters of the Student's t-distribution for the MH algorithm are calculated precisely enough by using the small data sampled at the beginning of the simulation. This observation may suggest that one can stop updating the parameters at some point of the simulation and use the same proposal density after that.

Acknowledgments

References

1. Mantegna, R., Stanley, H.E.: Introduction to Econophysics (Cambridge University Press)(1999).
2. Cont, R.: Empirical Properties of Asset Returns: Stylized Facts and Statistical Issues. Quantitative Finance 1 (2001) 223–236 (2001).
3. Engle, R. F.: Autoregressive Conditional Heteroskedasticity with Estimates of the Variance of the United Kingdom inflation. Econometrica 60, 987–1007 (1982)
4. Bollerslev, T.: Generalized Autoregressive Conditional Heteroskedasticity. Journal of Econometrics 31, 307–327 (1986)
5. Black, F.: The Pricing of Commodity Contracts. Journal of Financial Economics 3(1-2), 167–179 (1976)
6. Nelson, D.B.: Conditional Heteroskedasticity in Asset Returns: A New Approach. Econometrica 59, 347–370 (1991)
7. Glosten, L. R., Jaganathan, R. Runkle, D.E.: On the Relation Between the Expected Value and the Volatility of the Nominal Excess on Stocks. Journal of Finance 48, 1779–1801 (1993)
8. Ding, Z., Granger, C.W.J., Engle, R.F.: A Long Memory Property of Stock Market Returns and a New Model. Journal of Empirical Finance 1, 83–106 (1993)
9. Zakoïan, M.: Threshold Heteroscedastic Models. Journal of Economic Dynamics and Control 18, 931–955 (1994)
10. Engle, R.F., Ng, V.: Measuring and testing the impact of news on volatility. Journal of Finance 48, 1749–1778 (1993)
11. Sentana, E.: Quadratic ARCH models. Review of Economic Studies 62, 639–661 (1995)
12. Bauwens, L., Lubrano, M.: Bayesian inference on GARCH models using the Gibbs sampler. Econometrics Journal 1, c23–c46 (1998)
13. Kim. S., Shephard, N., Chib, S.: Stochastic volatility: Likelihood inference and comparison with ARCH models. Review of Economic Studies 65, 361–393 (1998)
14. Nakatsuma, T.: Bayesian analysis of ARMA-GARCH models: Markov chain sampling approach. Journal of Econometrics 95, 57–69 (2000)
15. Vrontos, I.D., Dellaportas, P., Politis, D.N.: Full Bayesian inference for GARCH and EGARCH models. Journal of Business and Economic Statistics 18, 187–198 (2000)

16. Mitsui, H., Watanabe, T.: Bayesian analysis of GARCH option pricing models. J. Japan Statist. Soc. (Japanese Issue) 33, 307–324 (2003)
17. Asai, M.: Comparison of MCMC Methods for Estimating GARCH Models. J. Japan Statist. Soc. 36, 199–212 (2006)
18. Takaishi, T.: Bayesian Estimation of GARCH model by Hybrid Monte Carlo. Proceedings of the 9th Joint Conference on Information Sciences 2006, CIEF-214 doi:10.2991/jcis.2006.159
19. Ardia. D.: Financial Risk Management with Bayesian Estimation of GARCH models. (Springer) (2008)
20. Takaishi, T.: An Adaptive Markov Chain Monte Carlo Method for GARCH Model. arXiv:0901.0992v1
21. Takaishi, T.: Bayesian Estimation of GARCH Model with an Adaptive Proposal Density. New Advances in Intelligent Decision Technologies, Studies in Computational Intelligence Volume 199, 635–643 (2009) doi:10.1007/978-3-642-00909-9_61
22. Takaishi, T.: Bayesian Inference on QGARCH Model Using the Adaptive Construction Scheme. Proceedings of 8th IEEE/ACIS International Conference on Computer and Information Science, 525–529 (2009) doi:10.1109/ICIS.2009.173
23. Metropolis, N. *et al.*: Equations of State Calculations by Fast Computing Machines. J. of Chem. Phys. 21, 1087–1091 (1953)
24. Hastings, W.K.: Monte Carlo Sampling Methods Using Markov Chains and Their Applications. Biometrika 57, 97–109 (1970)